REMARKS

Claims 1-25 are pending and stand rejected. Based on the following remarks, the Applicants respectfully request that the Examiner withdraw the rejections and pass the application on to issuance.

Claim Rejections – 35 USC §102: The Examiner rejected Claims 1-6, 9-14, 17, and 19-25 under §102 as being anticipated by USPN 6,490,624 issued to Sampson. Sampson is directed to session management in a stateless network such as the Internet. *See, e.g.,* Sampson, Title and Abstract. Sampson's system includes a number of access servers each of which acts as a gatekeeper for a protected server. Session information for a given client is stored in a session manager bound to an access server. In operation a client logs into an access server for a first protected server and then submits a request for a resource of a second protected server. The session manger for the access server determines whether the client has any authenticated sessions with any other access servers. If so, the client is permitted to access the resource of the second protected server without first logging in. *See* Sampson, Abstract.

Claim 1 is directed to a method for locating a resource and recites the following acts:

1. providing an interface having instructions to send association data;
2. identifying an identity service using the association data, the identity service managing resource data; and
3. locating the resource using the resource data.

It is noted that in a prior office action the Examiner admitted that Sampson failed to teach the second and third acts. Now, however, the Examiner contends:

- the act of providing is taught be Sampson, col. 9, lines 4-5;

2

- the act of identifying is taught by Sampson, col. 13, lines 5-67; and
- the act of locating is taught by Sampson, col. 14, lines 25-35.

As for the act of providing, Sampson col. 9, lines 4-6 read as follows: "As in Fig. 1, client 100 executes browser 101 and communicates with one or more Access Servers 104A, 104B directly or indirectly through network 102." Nothing in this passage even suggests "providing an interface having instructions to send association data" as recited by Claim 1. The passage merely describes the common use of a browser to communicate with a server over a network. There is no mention of the provision of an interface let alone the provision of an interface having instructions to send association data. Should the Examiner persist, the Applicants respectfully ask that he explain how this passage could be interpreted to teach the act of providing recited by Claim 1.

As for the act of identifying, Sampson col. 13, lines 5-67 are reproduced as follows:

> The Session Manager object takes the Session ID and performs checks on it. For example, as shown in block 532, the Session Manager object checks to determine whether the Session ID is recognized or known, by searching a local hash table of the Session Manager object to find the Session ID. If the Session ID is not found in the local hash table, then an internal error is generated, and a login page is returned to Client 100, as indicated by block 533. Thereafter, to gain access to the Protected Server, the Client 100 must provide valid username and password information through the login page to the Access Server. This prevents malicious users or processes from entering the system using an invalid Session ID.
>
> In block 534, the Session Manager object checks to determine whether the Session ID has been revoked. Revocation may occur, for example, because a session has terminated, or as a result of action by an administrator to intentionally revoke the session.
>
> In block 536, the Session Manager object also checks to determine whether an idle timeout or general timeout has occurred with respect to the current Session ID, by comparing the timeout values to a last access time value that indicates the last time that the client used an Access Server.

3

S/N: 10/085,927
Case: 100201141-1
Response to Office Action

Based upon the results of the checks of block 534 and block 536, a status code value is created and stored, as shown by block 538. The status code value indicates the status of the session associated with the Session ID. After the checks are complete, in block 540, the status code value is tested to determine whether a valid session exists. For example, the status code value may correspond to an Online state. In the Online state, the request of Client 100 is permitted to go through the system and the client may access the Protected Server and its resources, as indicated by block 542, in which access is granted.

Otherwise, based on the status code, access is refused, and a message is sent to the client that explains why the client is not permitted to access the requested resource, as shown by block 544 and block 546. The message may be in the form of an electronic document, such as an HTML page that can be displayed by Browser 101. The specific content of the electronic document may be established by the system administrator. In the case of an idle timeout event, the electronic document could comprise an HTML page stating, "Cannot Access Protected Resource (Session Idle Too Long)," or a similar message. The electronic document is created and sent to the client by the Runtime 406A, 406B upon receipt of the status code value from the Session Manager 420A, 420B.

In the preferred embodiment, each Session Manager object implements methods that carry out these and other functions, including:

put—Adds a new session to the database 450.

deleteSession—Deletes a session from the database.

updateSession—Updates the session.

get—Returns session information.

revokeUser—Revokes all sessions that are associated with a particular user, based on a user identifier value.

getNumberOfSessions—Returns a list of all sessions that are managed by the Session Manager, including information indicating the status of each session, such as Revoked, Expired, etc.

Nothing in this passage teaches or suggests "identifying an identity service using the association data, the identity service managing resource data" as recited by Claim 1. The first paragraph merely describes a session manager checking a session ID to determine whether it is known. The second paragraph describes the session manager

4

checking to determine if the session ID has been revoked. The third paragraph describes checking whether a timeout has occurred with respect to the session ID.

NONE of those checks involve using the session ID to identify an identity service that manages resource data.

The fourth and fifth paragraphs describe generating a status code value based on the outcome of the checks mentioned in the first three paragraphs. That status code value represents the status of a session associated with the session ID. That value is then tested to determine if a valid session exists. Based on the value of the status code value, access to a protected server is either granted or denied. Such is irrelevant to the requirements of Claim 1.

Consequently, neither the status code value nor the session ID is used to identify an identity service that manages resource data.

As for the act of locating, Sampson, col. 14, lines 25-35 are reproduced as follows:

> Client 100 then requests a protected resource from Protected
> Server 104B. Runtime 406B updates the Last Access Time value, and
> provides it to Session Manager 420A, which also updates its copy of the
> Last Access Time. Assume that Client 100 actively works with resources
> managed by Protected Server 104B for more than 15 minutes, and then
> returns to Access Server 104A to obtain one of its protected resources.
> Since the Last Access Time value is updated by Session Manager 420B
> each time Client 100 interacts with Access Server 104B, Session Manager
> 420A determines that Client 100 is active and may interact with Protected
> Server 104A to access its resources.

Nothing in this passage teaches or suggests "locating the resource using the resource data" as recited by Claim 1. The passage simply describes a client requesting a protected resource from a protected resource server and then updating a last access time value. The passage mentions nothing of the use of resource data, let alone the use of resource data to locate a resource.

For at least these reasons, Claim 1 is patentable over Sampson. Claims 2-4 are also patentable over Sampson due at least in part to their dependence from Claim 1.

5

Claim 5 is directed to a method for locating a resource for a user and recites the following acts:

1. providing an interface having instructions to send association data to two or more association services;
2. identifying from the two or more association services, an association service with which the user has established a relationship;
3. identifying an identity service using the association data sent to the identified association service, the identity service managing resource data; and
4. locating the resource using the resource data.

It is noted that in a prior office action the Examiner admitted that Sampson failed to teach the second and third acts. Now the Examiner contends:

- the act of providing is taught be Sampson, col. 9, lines 4-5;
- the act of identifying from the two or more association services is taught be Sampson, col. 10, lines 32-45;
- the act of identifying an identity service is taught by Sampson, col. 13, lines 5-67; and
- the act of locating is taught by Sampson, col. 14, lines 25-35.

As for the act of providing, Sampson col. 9, lines 4-6 read as follows: "As in Fig. 1, client 100 executes browser 101 and communicates with one or more Access Servers 104A, 104B directly or indirectly through network 102." Nothing in this passage even suggests "providing an interface having instructions to send association data to two or more association services" as recited by Claim 5. The passage merely describes the common use of a browser to communicate with a server over a network. There is no mention of the provision of an interface let alone the provision of an interface having instructions to send association data to two or more association services. Should the

6

Examiner persist, the Applicants respectfully ask that he explain how this passage could be interpreted to teach the act of providing recited by Claim 5.

As for the act of identifying from the two or more association services, Sampson, col. 10, lines 32-45 are reproduced as follows:

> The Session Managers may be organized in one or more clusters and there may be one Topology Mechanism for each cluster that keeps track of each of the Session Managers that are in that cluster. Normally the Topology Mechanism does not actively contact Session Managers, except in response to a request for information from a Session Manager, or in response to a Session Manager going offline.
>
> Session management in the system 400 is carried out with respect to sessions between clients such as client 100 and servers such as Protected Server 104, 112. Each session between a client and a server is represented by a set of session information. The session information preferably comprises: an initial session identifier value; an initial access time value; a last access time value; a user identifier value or key; a general timeout value; and an idle timeout value.

Nothing in this passage teaches or suggests "identifying from the two or more association services, an association service with which the user has established a relationship" as recited by Claim 5. The first paragraph simply notes that session managers may be organized in clusters and that a mechanism may be in place to track each cluster. There is no mention that the mechanism identifies a particular session manager with which a user has established a relationship.

The second paragraph describes session management between a client and a server where each session is represented by session information. The second paragraph lists potential types of data to be included in the session information. Nothing in the second paragraph teaches or suggests from two or more session managers a session manager with which a user has established a relationship.

As for the act of identifying an identity service, Sampson, col. 13 lines 5-67 were reproduced above with respect to Claim 1. The various paragraphs in that passage

<div align="center">7</div>

describe a session manager checking a session ID to determine whether it is known, checking to determine if the session ID has been revoked, and checking whether a timeout has occurred with respect to the session ID. NONE of those checks involve using the session ID to identify an identity service that manages resource data. The passage also describes generating a status code value based on the outcome of the checks. That status code value represents the status of a session associated with the session ID. That value is then tested to determine if a valid session exists. Based on the value of the status code value, access to a protected server is either granted or denied.

As clarified above with respect to Claim 1, that passage does not teach or suggest the use of association data to identify an identity service that manages resource data. As such that same passage fails to teach or suggest "identifying an identity service using the association data sent to the identified association service, the identity service managing resource data" as recited by Claim 5.

As for the act of locating the resource using the resource data, Sampson, col. 14, lines 25-35 were reproduced above with respect to Claim 1. Nothing in that passage teaches or suggests "locating the resource using the resource data" as recited by Claim 5. Instead, that passage describes a client requesting a protected resource from a protected resource server and then updating a last access time value. The passage mentions nothing of the use of resource data, let alone the use of resource data to locate a resource.

For at least these reasons, Claim 5 is patentable over Sampson.

Claim 6 is directed to a method, in a computer network, for locating a resource and recites the following acts:

1. providing a web page having instructions to request a web bug;
2. requesting the web bug sending a cookie and an URL for the web page;

3. saving the cookie and the URL for the web page as an entry in an association table;

4. querying, providing the URL for the web page, the association table for the cookie in the entry containing the URL;

5. identifying other entries in the association table containing the cookie;

6. identifying from those entries an entry containing an URL for an identification service, the identification service managing resource data; and

7. locating the resource using the resource data.

The Examiner contends:

- the acts of providing a web page, requesting a web bug, and saving the cookie are taught be Sampson, col. 10, "lines 10-4)" [it is assumed that the Examiner meant lines 10-40];

- the acts of querying, identifying other entries, and identifying from those entries, and locating are taught by Sampson, col. 14, lines 25-35.

As for the acts of providing a web page, requesting a web bug, and saving the cookie, Sampson col. 10, lines 5-46 read as follows:

> Topology Mechanism 440 is coupled to each of the Session Managers 420A, 420B and to the Logging Service 430. The Topology Mechanism 440 keeps track of all the Session Managers or replica that are in existence at any given time. To do this, the Topology Mechanism 440 stores information identifying each Session Manager that exists, and information identifying each Access Server that is associated with or bound to that Session Manager.

> In the preferred embodiment, Topology Mechanism 440 is implemented in the form of an object compliant with the Common Object Request Broker Architecture (CORBA) that can monitor other objects in the system. Its monitoring capabilities may be implemented based on Interceptor objects, which are part of the commercially available Visibroker ORB system. Interceptors are add-ons to existing objects. Logically, they

9

are located between a client to an object and the object itself. Interceptors allow a system to track connections and messages to objects.

In the Topology Mechanism 440, an Interceptor is instantiated each time a Session Manager 420, 420B binds to the Topology Mechanism. When a Session Manager object is started, it connects to the Topology Mechanism, registers itself, and sets its state. In response, the Topology Mechanism installs an Interceptor for that Session Manager object. If a session connection is dropped, the Topology Mechanism detects the drop and sets the state value of the Session Manager object to Down. These mechanisms are described further herein.

The Session Managers may be organized in one or more clusters and there may be one Topology Mechanism for each cluster that keeps track of each of the Session Managers that are in that cluster. Normally the Topology Mechanism does not actively contact Session Managers, except in response to a request for information from a Session Manager, or in response to a Session Manager going offline.

Session management in the system 400 is carried out with respect to sessions between clients such as client 100 and servers such as Protected Server 104, 112. Each session between a client and a server is represented by a set of session information. The session information preferably comprises: an initial session identifier value; an initial access time value; a last access time value; a user identifier value or key; a general timeout value; and an idle timeout value.

Nothing in the passage teaches or suggests the acts of providing a web page, requesting a web bug, and saving the cookie recited by Claim 6. The first paragraph (col. 10, lines 5-12) describes a topology mechanism that is responsible for tracking session managers. The second paragraph (col. 10, lines 13-21) further describes the topology mechanism as being implemented as an object that can monitor other objects and that is compliant with a particular architecture. The second paragraph also adds that the topology mechanism may be an interceptor object from a commercially available Visibroker ORB system. The third paragraph (col. 10, lines 22-31) describes that in the topology mechanism an interceptor is instantiated each time a session manager binds to the topology mechanism.

The fourth paragraph (col. 10, lines 32-38) simply notes that session managers may be organized in clusters and that a mechanism may be in place to track each cluster. The fifth paragraph (col. 10, lines 39-46) describes session management

10

between a client and a server where each session is represented by session information. The fifth paragraph lists potential types of data to be included in the session information.

Nothing in these paragraphs teaches, suggests, or even hints at a method that includes providing a web page having instructions to request a web bug, requesting the web bug sending a cookie and an URL for the web page, and saving the cookie and the URL for the web page as an entry in an association table as recited by Claim 6.

As for the acts of querying, identifying other entries, and identifying from those entries, and locating, Sampson, col. 14, lines 25-35 is reproduced as follows:

> Server 104B. Runtime 406B updates the Last Access Time value, and provides it to Session Manager 420A, which also updates its copy of the Last Access Time. Assume that Client 100 actively works with resources managed by Protected Server 104B for more than 15 minutes, and then returns to Access Server 104A to obtain one of its protected resources. Since the Last Access Time value is updated by Session Manager 420B each time Client 100 interacts with Access Server 104B, Session Manager 420A determines that Client 100 is active and may interact with Protected Server 104A to access its resources.

The passage simply describes a client requesting a protected resource from a protected resource server and then updating a last access time value. The passage fails to teach, suggest, or even hint at a method that includes identifying other entries in the association table containing the cookie, identifying from those entries an entry containing an URL for an identification service, the identification service managing resource data, and locating the resource using the resource data.

For at least these reasons Claim 6 is patentable over Sampson.

Claim 9 is directed to a computer readable medium having instructions for implementing the method of Claim 1. For at least the same reasons Claim 1 is patentable, so are Claim 9 and Claims 10-12 which depend from Claim 9.

<div align="center">11</div>

Claim 13 is directed to a computer readable medium having instructions for implementing the method of Claim 5. For at least the same reasons Claim 5 is patentable, so is Claim 13.

Claim 14 is direct to is directed to a computer readable medium having instructions for implementing the method of Claim 6. For at least the same reasons Claim 6 is patentable, so is Claim 14.

Claim 17 is direct to a system for locating a resource, and recites the following elements:

1. an association module operable to query an association service, supplying a session identifier, in order to identify an identity service managing resource data; and
2. an application operable to:
   a. provide an interface having instructions to send association data to the association service, the association data to contain a client identifier and a session identifier for the provided interface;
   b. acquire resource data from an identity service identified by a query from the association module; and
   c. locate the resource using the resource data.

In short, Claim 17 recites a system capable of implementing the method of Claim 1. For some mysterious reason, the Examiner rejected Claim 17 citing the same grounds used to reject Claim 5. For at least the same reasons Claim 1 is patentable, so is Claim 17 and Claim 18 which depends from Claim 17.

Claim 19 is directed to a document production system and recites the following elements:

12

1. an association module operable to query an association service, supplying a session identifier in order to identify an identity service managing resource data; and

2. a document production application operable to:
   a. provide an interface having content for sending association data containing a session identifier for the provided interface to an association service as well as content for displaying controls for selecting production options;
   b. acquire resource data from an identity service identifier identified by a query from the association module;
   c. locate and access a document management service using the resource data; and
   d. provide, for the interface, additional content for displaying controls for selecting a document managed by the document management service; and
   e. produce a document according to selections made through the interface.

Rejecting Claim 19, the Examiner makes the following unrelated explanation:

> As per claims 19, 20-25, Sampson et al teaches a system for locating a resource, comprising: an identity service operable to manage resource data; an association server operable to receive association data containing a client identifier and a session identifier, save the association data in an association table, and receive queries for the association table (See col. 10, lines 40-45); an association table interface in communication with the association server and operable, according to a received query, to access from the association table a session identifier for the identity service using a session identifier supplied with the query (See col. 11, lines 1-14); an association module operable to query, supplying a session identifier, the association service in order to identify the identity service; an application operable to: provide an interface having instructions to send association data to an association server, the association data to

13

contain a client identifier and a session identifier for the provided interface (See 9, lines 52-67); Furthermore, Sampson teaches acquiring resource data from the identity service identified by a query from the association module; and locate the resource using the resource data (See col. 14, lines 25-35).

It appears that the Examiner has not even read Claim 19 as the explanation for rejecting the Claim is solely focused on the elements of Claim 20 which differ substantially if not entirely from the elements of Claim 19. Inasmuch as the Examiner has failed to examine Claim 19, the rejection cannot stand.

Claim 20 is directed to a system for locating a resource and recites the following elements:

1. an identity service operable to manage resource data;
2. an association server operable to receive association data containing a client identifier and a session identifier, save the association data in an association table, and receive queries for the association table;
3. an association table interface in communication with the association server and operable, according to a received query, to access from the association table a session identifier for the identity service using a session identifier supplied with the query;
4. an association module operable to query, supplying a session identifier, the association service in order to identify the identity service;
5. an application operable to:
   a) provide an interface having instructions to send association data to an association server, the association data to contain a client identifier and a session identifier for the provided interface;
   b) acquire resource data from the identity service identified by a query from the association module; and
   c) locate the resource using the resource data.

The Examiner asserts that:

- the identity server and association server elements of Claim 20 are taught by Sampson, col. 10, lines 40-45;
- the association table interface element is taught by Sampson, col. 11, lines 1-14; and
- the association module and application elements are taught by Sampson, col. 9, lines 52-67 and col. 14, lines 25-35.

As for the identity server and association server elements Sampson, col. 10, lines 39-46 are reproduced as follows:

> Session management in the system 400 is carried out with respect to sessions between clients such as client 100 and servers such as Protected Server 104, 112. Each session between a client and a server is represented by a set of session information. The session information preferably comprises: an initial session identifier value; an initial access time value; a last access time value; a user identifier value or key; a general timeout value; and an idle timeout value.

As previously mentioned, this paragraph describes session management between a client and a server where each session is represented by session information. The passage lists potential types of data to be included in the session information. Nothing in this passage teaches, suggests or even hints at a system that includes an identity service operable to manage resource data, and an association server operable to receive association data containing a client identifier and a session identifier, save the association data in an association table, and receive queries for the association table.

With respect to the association table interface element, Sampson, col. 11, lines 1-16 are reproduced as follows:

<div align="center">15</div>

The session information may be stored in volatile memory or in a persistent storage medium. The session information may be retained only for a limited period of time, for example, one hour. Preferably, the session information is associated and stored in a session object that is managed using a CORBA-compliant Object Request Broker (ORB), such as the Inprise CORBA ORB. However, this method of representing the session information is not required, and any other form of information representation may be used. It is preferred to store session objects in memory for a pre-determined period of time, such as one hour, so that they do not need to be re-created if the user initiates a connection or session during that period after a period of idleness.

The general timeout value and the idle timeout values are pre-set at pre-determined values by the system administrator, and their use is described further herein.

This passage merely describes session information being stored in volatile memory. That session information may be kept for a limited period of time and is managed using a particular type of object request broker. The passage further describes that the time out values included in the session information are preset. The passage mentions nothing of an association table interface let alone "an association table interface in communication with the association server and operable, according to a received query, to access from the association table a session identifier for the identity service using a session identifier supplied with the query" as recited by Claim 20.

As for the association module and application elements. Sampson, col. 9, lines 52 through col. 10, line 4 and col. 14, lines 25-35 are reproduced as follows:

Database 450 maintains a list of sessions. All Session Managers 420A, 420B know the list of sessions. In one implementation, the list is kept in memory, and any change to a session is broadcast to all Session Managers. Alternatively, the list of sessions may be maintained in a database table. Database replication may be used to provide redundancy. Each Session Manager may be located in the same computer as the computer that hosts the database.

16

A Logging Service 430 may be coupled to each of the Session Managers 420A, 420B. The Logging Service 430 receives information about the actions taken by the Session Managers and records such information in one or more logs. If a session is removed from memory, an administrator can determine what happened to the session information by reviewed the logs. Preferably, Logging Service 430 is called to log exceptions; session creation; session revocation; session revocation by administrator; and session revocation due to idle timeout. Each log comprises a plurality of records. Each log record includes a session identifier and information identifying the client that caused the logged event.

Sampson, col. 9, lines 52 through col. 10, line 4.

Client 100 then requests a protected resource from Protected Server 104B. Runtime 406B updates the Last Access Time value, and provides it to Session Manager 420A, which also updates its copy of the Last Access Time. Assume that Client 100 actively works with resources managed by Protected Server 104B for more than 15 minutes, and then returns to Access Server 104A to obtain one of its protected resources. Since the Last Access Time value is updated by Session Manager 420B each time Client 100 interacts with Access Server 104B, Session Manager 420A determines that Client 100 is active and may interact with Protected Server 104A to access its resources.

Sampson, col. 14, lines 25-36.

The first passage (col. 9, lines 52 through col. 10, line 4) simply describes a database that maintains a list of sessions. That passage also describes a logging service that records actions taken be session managers in a log. The second passage simply describes a client requesting a protected resource from a protected resource server and then updating a last access time value.

Nothing in either of these passages teaches an association module or an application as those elements are recites in Claim 20.. More particularly the passages fail to teach or suggest an "association module operable to query, supplying a session identifier, the association service in order to identify the identity service." The passages fail to teach or suggest an "application operable to provide an interface having

17

instructions to send association data to an association server, the association data to contain a client identifier and a session identifier for the provided interface; acquire resource data from the identity service identified by a query from the association module; and locate the resource using the resource data."

For at least these reasons, Claim 20 is patentable over the cited references as is Claim 21 which depends from Claim 20.

Claim 22 is directed to a document production system and recites the following elements:

1. a document management service;
2. an identity service operable to manage resource data for locating and accessing the document management service;
3. an association server operable to receive association data containing a client identifier and a session identifier, save the association data in an association table, and receive queries for the association table;
4. an association table interface in communication with the association server and operable, according to a received query, to access from the association table a session identifier for the identity service using the session identifier supplied with the query;
5. an association module operable to query, supplying a session identifier, the association service in order to identify the identity service;
6. a document production application operable to:
   a. provide an interface having content for sending association data containing a client identifier and a session identifier for the provided interface to an association service as well as content for displaying controls for selecting production options;
   b. acquire resource data from an identity service using the session identifier for the identity service identified by a query from the association module;

c. locate and access the document management service using the resource data;

d. provide, for the interface, additional content for displaying controls for selecting a document managed by the document management service; and

e. produce a document according to selections made through the interface.

The Examiner bundles the explanation for the rejection of Claim 22 with the explanation or Claim 20. ONCE AGAIN, the Examiner is reminded that explanation fails to address specific elements of Claim 22 which are different than the elements of Claim 20. it has become clear that the Examiner has failed to Examine Claim 22. For at least this reason, the rejection of Claim 22 cannot stand. Furthermore, the system of Claim 22 recites elements for implementing the method of Claim 7. For at least the same reasons Claim 7 is patentable so are Claim 22 and Claim 23 which depends from Claim 22.

Claim 24 is directed to system for implementing the method of Claim 1. For at least the same reasons Claim 1 is patentable, so is Claim 24.

Claim 25 is directed to a system for implementing the method of Claim 7. For at least the same reasons Claim 7 is patentable, so is Claim 25.

Claim Rejections – 35 USC §103: The Examiner rejected Claims 4, 7-8, 15-16, and 18 under §103 as being unpatentable over Sampson in view of US Pub 2004/0015580 to Lu.

Claim 4 depends from Claim 1. For at least the same reasons Claim 1 is patentable, so is Claim 4.

Claim 7 is directed to a method for producing an electronic document and

19

S/N: 10/085,927
Case: 100201141-1
Response to Office Action

recites the following acts:

1. generating, upon request from a user, a web page having content for requesting a web bug from an association service as well as content for displaying controls for selecting production options;
2. querying the association service to identify an identity service with which the user is registered providing an URL for the generated web page;
3. obtaining the user's resource data from the identified identity service;
4. locating and accessing a document management service using the resource data;
5. providing additional content for the web page for displaying controls for selecting a document managed by the document management service; and
6. producing a document according to selections made through the web page.

The Examiner makes the following assertions:

1. the acts of generating, querying, and obtaining are taught by Sampson, col. 10, lines 40-45; and
2. the acts of locating, providing, and producing are not taught by Sampson, but are taught by Lu, paragraph [0064].

As for the acts of generating, querying, and obtaining, Sampson, col. 10, lines 39-46 are reproduced as follows:

> Session management in the system 400 is carried out with respect to sessions between clients such as client 100 and servers such as Protected Server 104, 112. Each session between a client and a server is represented by a set of session information. The session information preferably comprises: an initial session identifier value; an initial access time value; a last access time value; a user identifier value or key; a general timeout value; and an idle timeout value.

As previously mentioned, this paragraph describes session management between a client and a server where each session is represented by session information. The passage lists potential types of data to be included in the session information. Nothing in this passage teaches, suggests or even hints at a system that includes an identity service operable to manage resource data, and an association server operable to receive association data containing a client identifier and a session identifier, save the association data in an association table, and receive queries for the association table.

This passage fails to teach, suggest, or even hint at a method that includes the acts of generating, querying, and obtaining recited by Claim 7. More particularly, the passage mentions nothing of generating a web page let alone generating "a web page having content for requesting a web bug from an association service as well as content for displaying controls for selecting production options." The passage is completely unrelated to "querying the association service to identify an identity service with which the user is registered providing an URL for the generated web page." Furthermore, the passage has no relevance to "obtaining the user's resource data from the identified identity service."

As for the acts of locating, providing, and producing, Lu, paragraph [0064] is reproduced as follows:

> [0064] The illustration in FIG. 5 shows, at a high level, how the invention operates. The visitor makes a web page request in step (1) by typing in a URL into a browser program operating on the client node 36. The URL has a domain (such as amazon.com) that points it toward a particular web server 30 located on the Internet. That web server is the device on which the web site is stored. The web site is constructed using a html or JavaScript code including the original web page code (including text and images), data mining code, and additional cookie processing code supplied by the web tracking provider that performs the functions described in more detail below to establish and process a cookie right on the client node without additional interaction with the web tracking provider..

The relevance of this paragraph is highly suspect. It merely describes browsing to a particular URL referencing a web site stored by a web server. The web site includes data mining code and additional cookie processing code supplied by a web tracking provider. The cookie processing code processes a cookie on a client node without interacting with the web tracking provider.

Nothing in the paragraph teaches, suggest, or even hints at a method that includes the acts of locating, providing, and producing recited by Claim 7.

For at least these reasons, Claim 7 is patentable over Sampson and Lu as is Claim 8 which depends from Claim 7

Claim 15 is directed to a computer readable medium having instructions for implementing the method of Claim 7. For at least the same reasons Claim 7 is patentable, so are Claim 15 and Claim 16 which depends from Claim 15.

Claim 18 depends from Claim 17. For at least the same reasons Claim 17 is patentable, so is Claim 18.

Conclusion: In view of the foregoing remarks, the Applicant respectfully submits that the pending claims are in condition for allowance. Consequently, early and favorable action allowing these claims and passing the application to issue is earnestly solicited. The foregoing is believed to be a complete response to the outstanding Office Action.

Respectfully submitted,

Gregory Eugene Perkins, et al.

By _____

Jack H. McKinney

Reg. No. 45,685

January 19, 2006

23